

EIE

Escuela de
Ingeniería Eléctrica

Universidad de Costa Rica

Facultad de Ingeniería
Escuela de Ingeniería Eléctrica



UNIVERSIDAD DE
COSTA RICA

IE-0117

Programación Bajo Plataformas Abiertas

Laboratorio #5

Fernando Jiménez Ureña B74020

II ciclo
Noviembre 2021

1. Resumen

En el presente laboratorio, se estudian y se implementan las funciones en C, aplicándolas en diferentes programas. Entre los programas se encuentran los Métodos de Ordenamiento (Quick Sort y Bubble Sort), Resolución de un Laberinto y la Aplicación de un Filtro Borroso a una fotografía. Además se utilizarán recursos como son el Git de Ingeniería Eléctrica para el control de versiones de los programas y de Doxygen para la documentación de los mismos.

2. Nota Teórica

Es importante definir qué son las funciones en C. Las funciones son entidades que dado un conjunto de datos (los parámetros), se les encarga realizar una tarea muy concreta y se espera hasta obtener el resultado. Lo idóneo es dividir tareas complejas en porciones más simples que se implementan como funciones. La división y agrupación de tareas en funciones es uno de los aspectos más importantes en el diseño de un programa. [1] Cabe destacar que en C, todas las funciones están al mismo nivel, esto quiere decir que no se puede definir una función dentro de otra función.

El ordenamiento de Burbuja (*BubbleSort*) consiste en un sencillo algoritmo para poder ordenar los arreglos de un vector de forma ascendente o descendente. Sin embargo se considera que es uno de los métodos más ineficientes debido a que hace múltiples pasadas a lo largo de una lista. Compara los ítems adyacentes e intercambia los que no están en orden. Cada pasada a lo largo de la lista ubica el siguiente valor más grande en su lugar apropiado, por lo que en un arreglo de números grande, el programa puede llegar a tardar en ejecutarse. [2]

Por otro lado, el otro método de ordenamiento de datos denominado *QuickSort*, es uno de los métodos de ordenamiento más rápido que existe, ya que permite en ordenar los elementos de un arreglo mediante un algoritmo que se basa en orden n elementos en un tiempo proporcional [3]

La biblioteca CCV que se utiliza en el tercer problema del laboratorio, es un tipo de biblioteca que se denomina librería de visión por computador. Una librería estática es un archivo de objeto compilado que contiene todos los símbolos requeridos por el programa principal para operar (funciones, variables, etc.) en lugar de tener que extraer entidades separadas. El compilador no carga las bibliotecas estáticas en tiempo de ejecución; solo es necesario cargar el archivo ejecutable [4].

Mientras tanto, una biblioteca dinámica consiste en una colección de archivos u objetos a los que se les hace referencia en el momento de la compilación para proporcionar información ejecutable sobre cómo se usarán finalmente pero no se usan hasta el momento de la ejecución. En otras palabras, estos objetos están vinculados dinámicamente a ejecutables que los utilizan.[4]

3. Análisis de Resultados

3.1. Capicúa

En este problema se desarrolló un programa que determina si un número es Capicúa o no. Un número Capicúa es aquel que puede leerse de igual manera ya sea de izquierda a derecha, así como de derecha a izquierda. En la Figura 1. puede apreciarse el diagrama de flujo del código implementado y en la Figura 2. la compilación y funcionamiento del mismo.

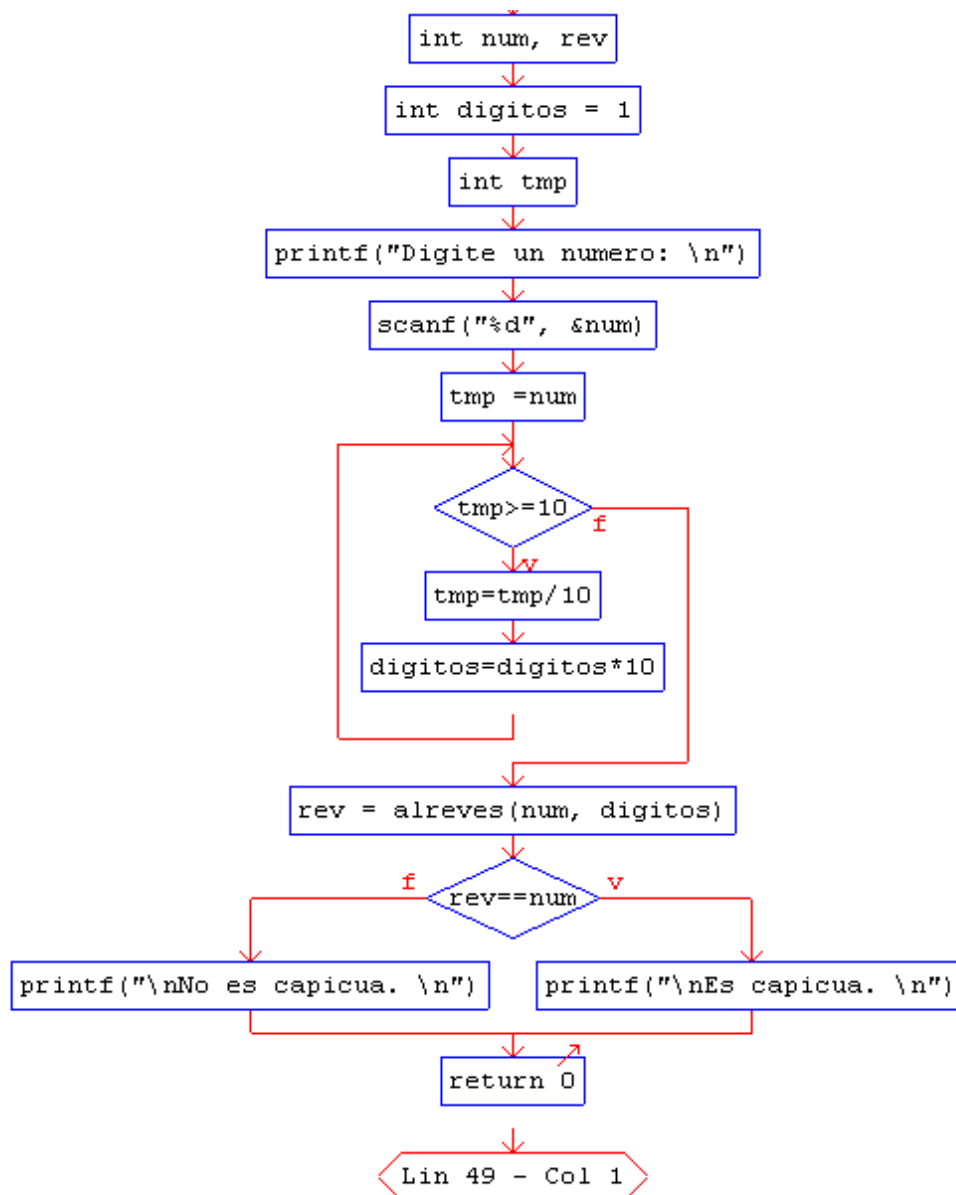


Figura 1: Diagrama de Flujo del código

```
fer9828@LAPTOP-K1I7TV04:/mnt/c/Users/jimen/OneDrive/Escritorio/Progra/C/Plataformas 2S - 2021/Laboratorio_5$ gcc -o exe Capicua.c
fer9828@LAPTOP-K1I7TV04:/mnt/c/Users/jimen/OneDrive/Escritorio/Progra/C/Plataformas 2S - 2021/Laboratorio_5$ ./exe
Digite un numero:
121

Es capicua.
fer9828@LAPTOP-K1I7TV04:/mnt/c/Users/jimen/OneDrive/Escritorio/Progra/C/Plataformas 2S - 2021/Laboratorio_5$ ./exe
Digite un numero:
156

No es capicua.
fer9828@LAPTOP-K1I7TV04:/mnt/c/Users/jimen/OneDrive/Escritorio/Progra/C/Plataformas 2S - 2021/Laboratorio_5$
```

Figura 2: Capicua corriendo en la Terminal

3.2. Métodos de Ordenamiento

En este problema, se implementaron los algoritmos de los métodos de ordenamiento *Quick-Sort* y *BubbleSort* descritos en la Nota Teórica. Se puede apreciar en la Figura 3. el diagrama de flujo del código implementado para la solución del problema. Además, en la Figura 4., se puede apreciar el código siendo compilado en la terminal. Cabe destacar que el código fuente se encuentra en el repositorio del GIT.

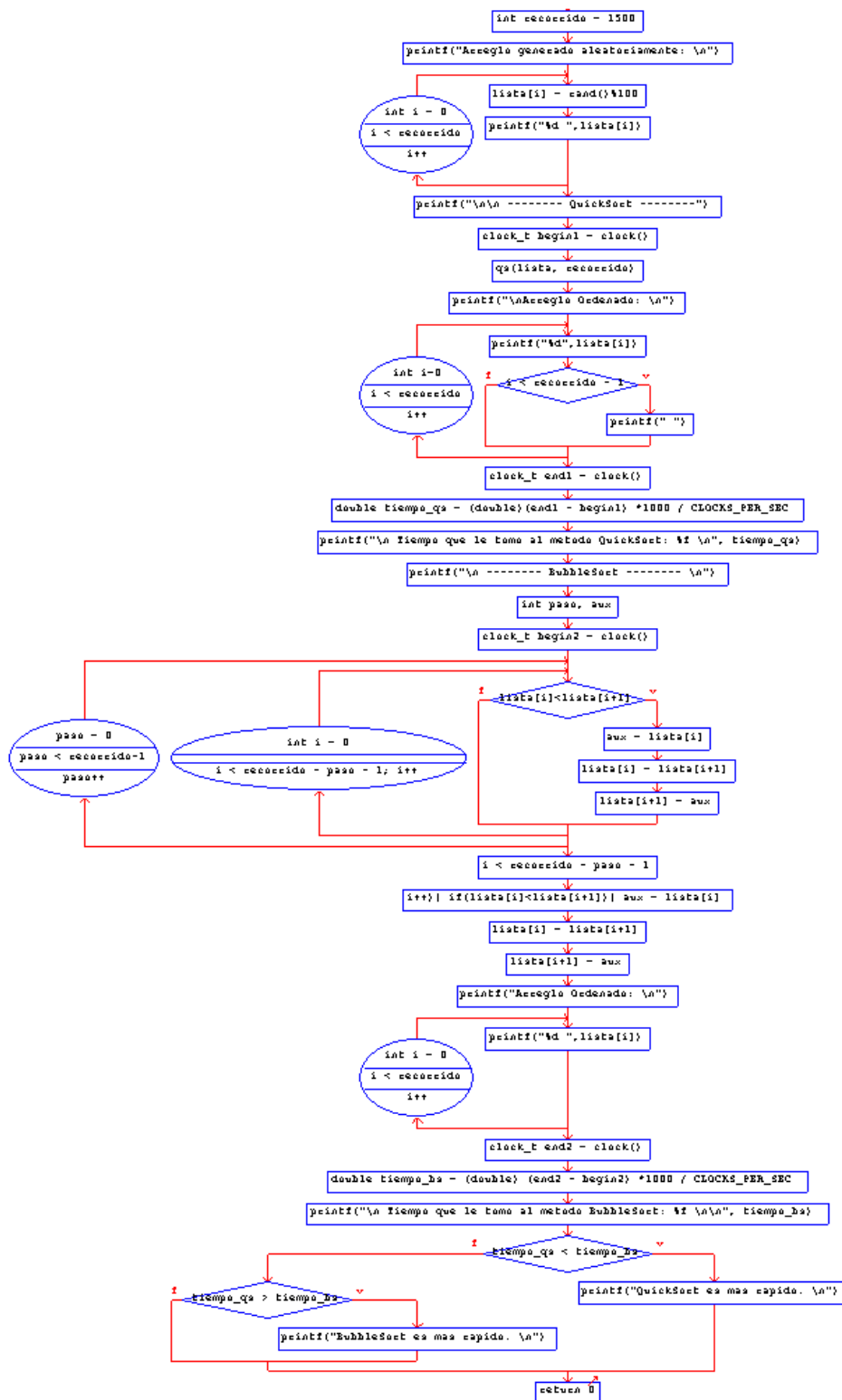


Figura 3: Diagrama de Flujo del código

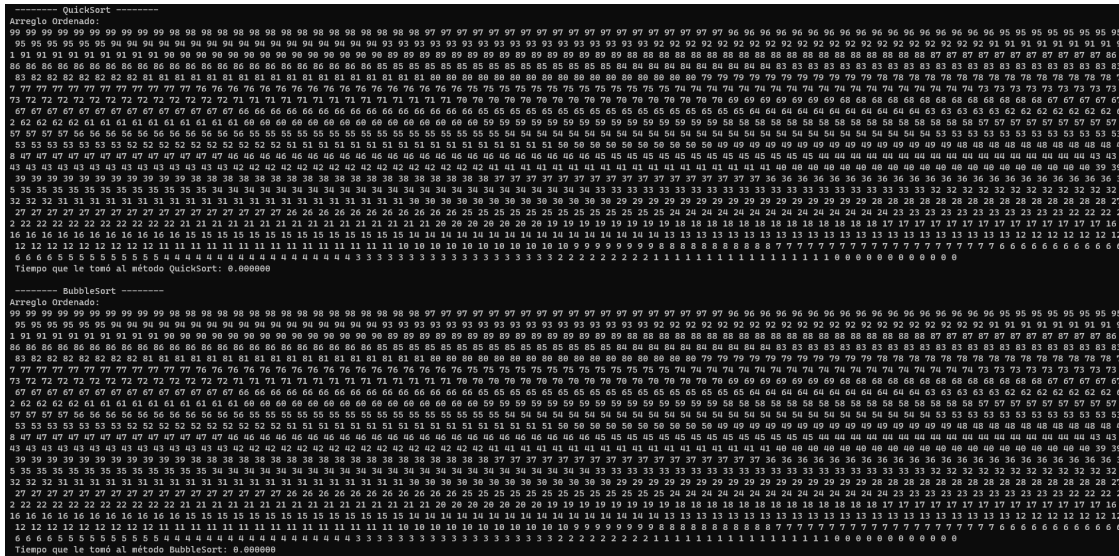


Figura 4: Métodos de Ordenamiento corriendo en la Terminal

3.3. Solución de un Laberinto

Para este problema, no se obtuvieron los resultados deseados. Luego de implementar una amplia línea de código, el programa no logró leer bien el archivo `laberinto.txt`. Esto causó que no se pudiese desarrollar correctamente la solución. Como se puede apreciar en la Figura 5., el programa no lograba detectar de manera completa el laberinto planteado en el problema.

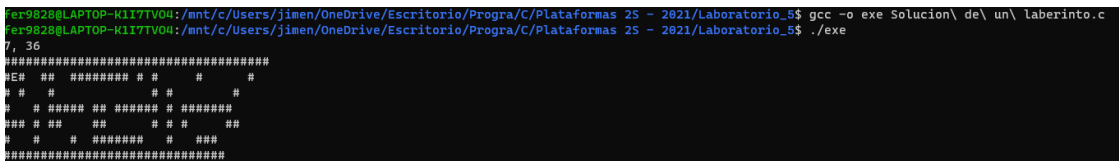


Figura 5: Laberinto no detectado

4. Conclusiones y Recomendaciones

En el presente laboratorio se estudiaron y se hizo uso de las funciones en C. En los diferentes programas fue de bastante utilidad ya que se pudo tener un código más ordenado y el poder dividir la funcionalidad del programa en diferentes partes es una buena práctica de programación ya que permite una mayor libertad a la hora de realizar cambios y mejoras en los programas.

5. Referencias

- [1] Capítulo 4. Las funciones en C. It.uc3m.es. (2021). Retrieved 8 November 2021, from http://www.it.uc3m.es/pbasanta/asng/course_notes/functions_e.html.
- [2] 5.7. El ordenamiento burbuja — Solución de problemas con algoritmos y estructuras de datos. Runestone.academy. (2021). Retrieved 9 November 2021, from <https://runestone.academy/runestone/s>
- [3] 5.12. El ordenamiento rápido — Solución de problemas con algoritmos y estructuras de datos. Runestone.academy. (2021). Retrieved 9 November 2021, from <https://runestone.academy/runestone/s>
- [4] L. Cheparro. "Static and Dynamic Libraries in C language". recuperado de <https://medium.com/@luisch-medium-com-luischapparoc-dynamic-libraries-in-c-96a989848476>